**International Journal of Information Retrieval Research**
Volume 13 • Issue 1

# Effective Information Retrieval Framework for Twitter Data Analytics

Ravindra Kumar Singh, National Institute of Technology, Jalandhar, India*
https://orcid.org/0000-0003-1142-1954

## ABSTRACT

The widespread adoption of opinion mining and sentiment analysis in higher cognitive processes encourages the need for real time processing of social media data to capture the insights about user's sentiment polarity, user's opinions, and current trends. In recent years, lots of studies were conducted around the processing of data to achieve higher accuracy. But reducing the time of processing still remained challenging. Later, big data technologies came into existence to solve these challenges but those have its own set of complexities along with having hardware deadweight on the system. The contribution of this article is to touch upon mentioned challenges by presenting a climbable, quick and fault tolerant framework to process real-time data to extract hidden insights. This framework is versatile enough to support batch processing along with real time data streams in parallel and distributed environment. Experimental analysis of proposed framework on twitter posts concludes it as quicker, robust, fault tolerant, and comparatively more accurate with traditional approaches.

## KEYWORDS

Cache Management, Data Processing Framework, Message Broker, MongoDB, Parallel Processing, Python-dash, Real Time Analytics, Redis, Social Media Analytics, Visualization

## 1. INTRODUCTION

Social media has evolved with the phenomena that each individual in this virtual world is a social player and love to discuss their viewpoints on different domains (Zeng et al., 2010), such as politics and journalism (Stieglitz & Dang-Xuan, 2013), business (Beier & Wagner, 2016), sports and entertainment (Shen et al., 2016), science and technologies (Baars & Kemper, 2008), etc. on public forums day in day out without bearing any cost of membership. The most lucrative virtue of social media is its real time wide range of acceptableness of user's posts in multiple languages across the domains about various events, situations, feelings, opinion, day to day thoughts, feedback, shopping experiences, best wishes, knowledge sharing, advice, wish lists, future plans or anything out of the imaginations. These posts may be in form of text messages, images, videos, emoji, maps etc and accepts the reactions of other users on it as well as extending the features to be shared on multiple

DOI: 10.4018/IJIRR.325798 *Corresponding Author

such portals to attain more and more engagements. It leads the wide acceptability and exponentially grown popularity among the users, therefore produces immense quantity of data (Azab & ElSherif, 2018). Social media platforms are capturing the associated metadata along with users post to better define the matter of the post.

According to a survey most of the online users invest their couple of hours in online activities like social media networking, shopping, banking, reading, coding, playing, watching videos, etc. This increasing interest of users in online activities accelerated the growth of online generated data up-to 2.5 quintillion bytes per day (BSA, 2015), According to a study of IBM, approximately 35 zettabytes of data would be generated annually by fall of 2020 (2012). Besides, in every second data are produced continuously; 34,722 Likes on Facebook, about 571 new websites, and almost 175 million tweets, billions of transaction logs, etc, accumulated all these data is known as Social Media Big Data (Lynn et al., 2015). This growing set of user generated data brings the opportunities for understanding and analyzing several aspects, patterns, trends and sentiments (Kim et al., 2016) and extract useful insights from it to take the edge in this competitive world (Oh et al., 2016).

Social media platforms have been established in recent years as sources of trends, feedback, debate and sentiments on across the domain including politics (Khatua et al., 2015), journalism, business (Kurniawati et al., 2013), entertainment, sports, etc. Business leaders across the domains are curious enough to use the real time insights from social media data in their policy and decision making, moreover to extend their support to gain better customer satisfaction (Nulty et al., 2016). Therefore "Social Media Analytics" coined as a special term and all these activities started from collecting posts to storing, analyzing till insight extraction and visualization fall under its umbrella (Stieglitz et al., 2014). All these factors made social media a separate world in itself among the internet domain, considerably influencing both academic and industrial outlooks (Batrinca, 2015).

## 2. BACKGROUND

Social media analytics have gained foremost attention (Tsantarliotis & Pitoura, 2017) since last couple of years in researcher's world for sentiment analysis (Kane et al., 2014) and opinion mining (Maynard et al., 2012). Researchers are more focused on real-time data (Jose & Chooralil, 2015), by using streaming application programming interface (API) of social media platforms like twitter (Tweepy, ), facebook, reddit, etc. to grasp the current trends and public opinions. Social media networking portals are operating on ideology of web 2.0 that defines that data would be created and updated by the users in collaborative manner rather than just being published by individuals who owns it (Kaplan & Haenlein, 2010). Therefore social media data are vague and not bound by any rule or specified formats (Hogenboom et al., 2011), so the research on this data arises the need of some framework to better define the steps of processing and could minimize the complexities in the research.

### 2.1 Social Media Analytics Framework

There are few frameworks to demonstrate and build a typical basis for conducting social media analytics. It is a multi step journey that contains data collection, data processing, data analysis and lands up with presentation and visualization of the outcome. A better flow is defined in the CUP framework (Fan & Gordon, 2014) which defines it in 3 phases-

- Capture is the process of data aggregation and applying pre-processing on it to get the better shape to the data.
- Understand is the process of applying various analytics per the use case on the captured data post handling noisy and skewed data to attain the insights of the analytics.
- Present is the process of evaluating and summarizing the findings.

However recently Chong Oh, et. al, suggested to feature an extra layer Identify for better identification of data sources at the beginning of the framework and presented the Modified CUP SMA Methodological Framework (2015) as given below in Figure 1-

CUP SMA Methodological Framework gained a wide range of attention and a lot of the researchers implemented that framework to pursue their research and to gauge its performance. Although this framework is well explained how to use and maximize the performance but not well exposed in terms of implementation, if we try to break it down into components, below would be usable components-

- Data / Streaming API: to capture the data
- Database: to store the collected data and analytics result.
- Pre-Processing Block: to apply the pre-processing on data to transform it into a better shape.
- Processing Block: to applying varied analytics per our use case and extract the useful insights.
- Visualization Unit: to plot and give visual sense to the extracted insights.

And based on that we could illustrate its flow for better understanding, Figure 2 is a simple illustration of this framework-

## 2.2 Big Data Technologies in Social Media Analytics

Discussions of social media data are commonly found in publications on big data and social media researchers frequently refer to the big data literature (Cao et al., 2015). This has been called social big data (Guellil & Boukhalfa, 2015) or social media big data (Lynn et al., 2015). Big data technologies are very scalable and fault torrent in nature. Maria Karanasou, et, al. presented a design and implementation of a real-time system architecture in Storm, which contains the feature extraction from tweets, using both morphological features and semantic information and do classifications. It could be scaled up well with respect to input data size and data arrival rate. Apart from that this architecture supports offline mode also as well as online mode of processing (Karanasou et al., 2016). Babak Yadranjiaghdam, et. al, developed an analytical framework, with the ability of in-memory processing to extract and analyze structured and unstructured Twitter data. This framework includes data ingestion, stream processing, and data visualization components, Apache Kafka is used to perform data ingestion task while Spark makes it possible to perform sophisticated data processing and machine learning algorithms in real time (Yadranjiaghdam et al., 2017).

**Figure 1. The modified CUP SMA methodological framework**

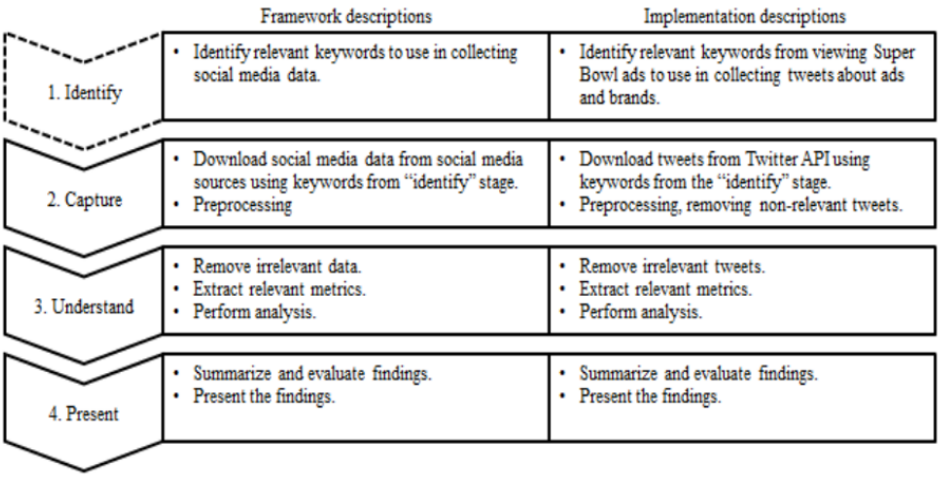
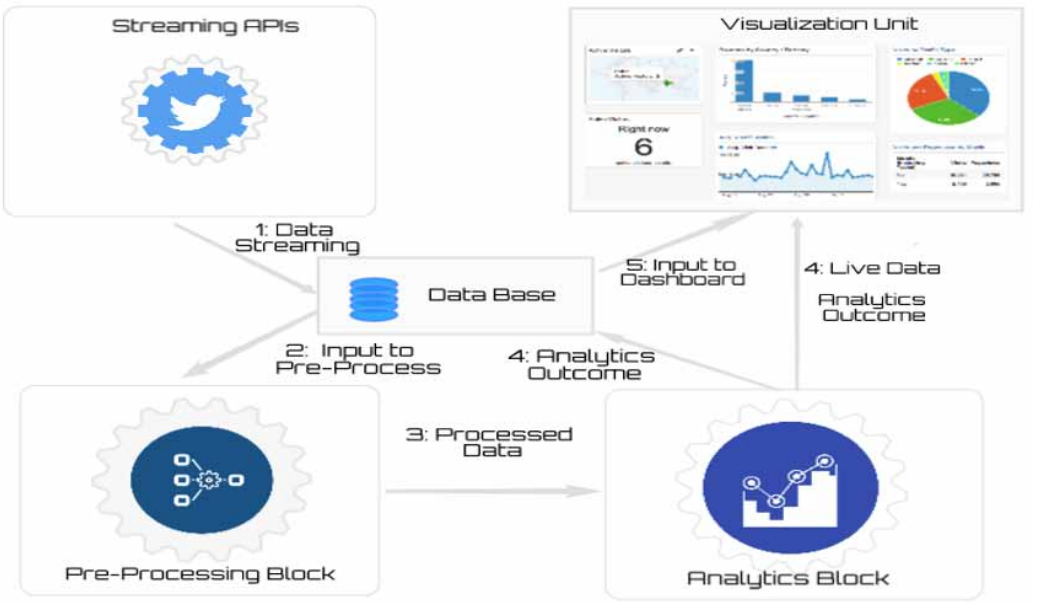


**Figure 2. Process flow diagram of system using CUP SMA methodological framework**




## 2.3 Current Challenges in Social Media Analytics

Despite having revolution in social media analytics, researchers still suffer to process the data in an effective manner to gain the analysis insights in relatively less time. Choosing and tuning a suitable machine learning algorithms and post tuning processing of data requires high configuration machines, which become hurdle in almost all the research in terms to achieve higher throughput (Saha et al., 2017). Generally researchers opt for big data technologies to scale out the hardware capabilities but it costs additional complexity on the system in terms of technological complexity, hardware overheads and extended time of the research. Most of such research or mid side companies don't have enough data to practice the big data implementation for processing real time data, it could be better managed by a good parallel processing framework (Lee, 2018) that could be easy to implement, fast enough to serve the purpose, robust and easy to scale out without bearing the challenges of big data implementation (Cao et al., 2015).

This research is not concerned about the sentiment prediction techniques or research on any particular domain but to maximize the utilization of resources by using better framework to achieve reasonably better accuracy with lower latency rate. Therefore this research is presenting a parallel processing framework to effectively process the social media posts and measuring its performance in term of execution time, cpu utilization and memory utilization.

## 3. PROPOSED FRAMEWORK

In this inter-connected internet era, information has become the precious most asset, even a single bit of data could influence a decision and could disrupt the business. Now a days the volume of such data is growing exponentially and the hidden secret inside these data points (insights) are worth of millions. So the topic comes to the corner that storing and driving useful insights from these data are very high in demand. The process to extract meaningful information from the raw data to drive a conclusion is known as analytics and there are lots of machine learning algorithms to deal with it, but still there are many challenges to be taken care-

- Handling high volume of data
- Integrating high speed continuous data
- Real time analytic with low latency

Big Data technologies came into existence to handle these challenges, and Storm (Karanasou et al., 2016) and Spark (Yadranjiaghdam et al., 2017) based frameworks gained most of the attraction (Chebbi, 2015), sometimes it costs additional overhead on computations and involves more complexity in the process (Younas, 2019) that could be better managed by implementing an efficient parallel processing framework. This research aims to provide an efficient, fast, robust framework to handle social media posts. Its bench-marking was done on twitter posts, but it could be extended to other corpus of data as well, especially on data having attached images, urls and other user's specific details.

## 3.1 Framework Layout

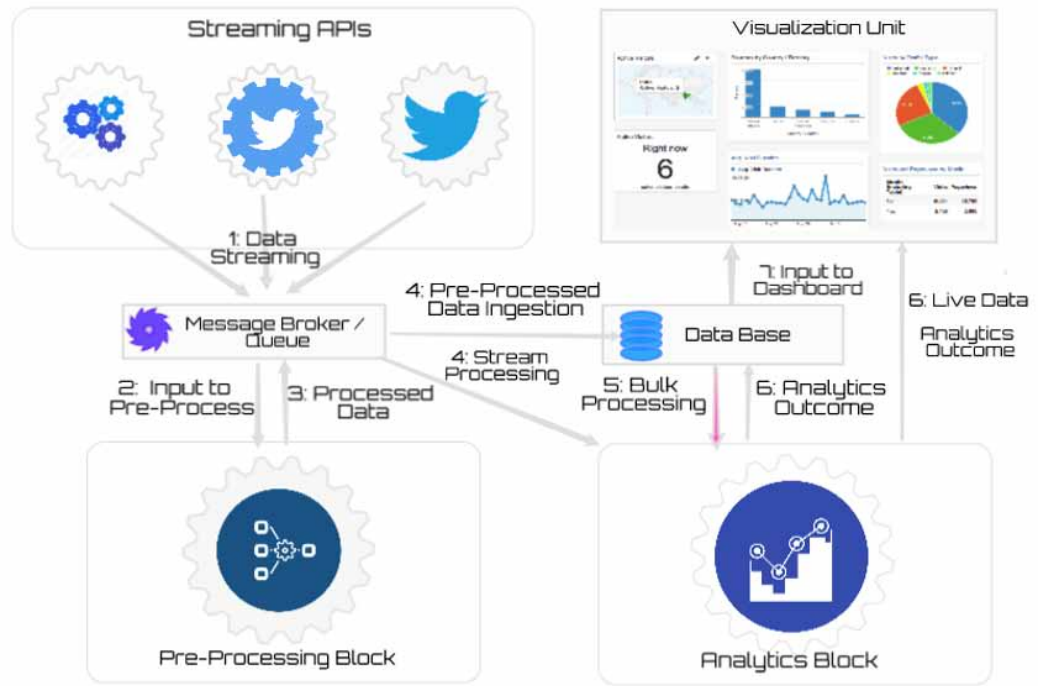
### *3.1.1 Basis Layout of the Framework*

Basis Layout of proposed Social Media Processing Framework is given below in Figure 3-

This framework comprises 6 major components -

- Streaming and Search API
- Message Broker
- Pre-Processing Block
- Database
- Analytics Block
- Visualization Unit

**Figure 3. Basic layout of the framework**



### 3.1.2 Components Description and Use Cases

#### 3.1.2.1 Streaming and Search API

The source of streaming and batch data, it may be some APIs provided by the social media providers, some mechanism to capture/scrap the data or any such source that could provide real time data.

#### 3.1.2.2 Message Broker

It helps to reduce the burden of IO operations during the analytics process, generally message brokers operates on memory itself so they are quite fast and capable to handle parallel requests, message brokers are ingesting data in permanent store as well for maintaining the reliability and recoverability of data. In this framework it would be leveraged to handle the bottleneck in the processing flow like storing streaming data, pre-processing and caching of re-utilizable cache data. Below are few use cases of message brokers in our design are given data-

- Implementing buffer between streaming API and database
- Enabling framework to perform parallel pre-processing on real time data to boost the performance by reducing the IO operations.
- Real time data sharing to the analytics block to avoid any possible delay after getting data from streaming APIs.
- Migrating the risk of data loss by storing the data parallely in database.
- It could be utilized as a cache to store important / reutilizable data in memory to make transactions fast without any IO interference.

#### 3.1.2.3 Pre-Processing Block

To transform the streaming data in well formatted data required by the analytics block. Pre-processing of data is a time consuming process, so it is designed for distributed parallel processing. Below are the use cases of separating pre-processing blocks from analytics block-

- To re-utilize the processed data
- Pre-processing is a time consuming task, so it could be distributed to run parallely. It is horizontally scalable.
- It would utilize the already processed piece of data in the form of cache using in-memory high speed data.

#### 3.1.2.4 Database

To store the incoming data as well as pre-processed well formatted intermediate data that could be utilized further, either for bulk analytics processing or to just to have a backup copy. Databases also preserve the final outcome of analytics that could be utilized by visualization block.

Below are few pretty use cases of database in this design-

- To store data for future use and to ensure against data loss and avoid re-processing of earlier processed data.
- It is storing pre-processed format of data to reduce the processing overhead and to ensure quick response.
- It would be required to run batch processing.
- Analytics outcome would be preserved and could be utilized in future.

#### *3.1.2.5 Analytics Block*

To extract the insights from pre-processed form of data having all required features. It would utilize few machine learning techniques to achieve the target. Finalizing better algorithm and its model training is out of the scope of this section; here data would be processed through already tuned model.

#### *3.1.2.6 Visualization*

To give the shape of analyzed data. Visuals make it easy to understand and explain that data, pattern and final conclusions, data is picked from the data stores and got displayed on some sort of dashboards to enable the user to have real time data monitoring. There are lots of charts, graphs and other components to give shape of the data that enables user to quickly grasp the information and see the real time transformation in data. These visuals on framework would be based on the project's use cases, and could be customized as per the requirements.

## 3.2 Open Source Implementation of the Framework

It's been observed a notably increment in the usage of user generated data (online social media) by individuals, countries and organizations for analysis, strategies and decision making purposes in the last few years but processing and visualization of these data by utilizing open source tools became the real concern. This research is going to represent an open source framework to get it all done in the best optimized way.

### *3.2.1 Streaming and Search APIs*

Almost all user portals (social media networking sites) are exposing their APIs for the individuals use, few are free publicly available having some limitation on its usage and on the other hand there are privileged services to access the data points. Suitable APIs and their plans could be selected based on the project requirements. Below are few such social media APIs-

- Twitter API
- Facebook Live API
- LinkedIn API
- Reddit API
- Wikipedia

The framework is tested for all above mentioned apis but it's bench-marking are done on Twitter's paid GNIP APIs to handle high speed streaming data.

### *3.2.2 Message Broker*

Below are the most popular open source message brokers that could be utilized as a buffer to the incoming data-

- Redis
- Apache ActiveMQ
- Apache Kafka
- RabbitMQ

In this implementation Redis (memory based database for buffer management & used as cache for critical information) is being used as message broker as well as the cache management to improve the performance. Although above mentioned all message brokers could be integrated in this framework, but Redis's ease of use, fast, efficient and multiple data type support makes the most suitable candidate for this purpose.

### 3.2.3 Pre-Processing Block

Pre-processing block is splitted from main processing unit in this framework to get the most of the reusability advantages, such as utilizing already processed data, etc. As being the most time consuming process, this could be parallely scaled as per the use case. One of the most ignorant and important facts on classification models is that a simple model on well featured data is always better than running a perfect model on noisy data, so this framework is more concerned of featuring the data well. The behavior and interests of the users are the most prominent keys to understand the importance of any individual and its post on the overall network (Zhang et al., 2016), this fact urges to include authors profile information too along with post contact, shared urls and attached images. Detailed list of parameters along with its processing steps are given below in Figure 4-

So this research concludes the overall sentiment of a post by using below equation (eq 1)-

$$Tweet_{Sentiment} = \left(Content_{Sentiment} + Image_{Sentiment} + Urls_{Sentiments}\right) \times User's_{Impact} \quad (1)$$

Apart from that we are leveraging the use of Cython and a bit of Numba library to optimize the pre-processing block.

#### 3.2.3.1 Parallel Processing

Pre-processing block could be scaled up by parallel processing arrangement, by processing multiple data-points simultaneously. Representation of parallel processing of tweets is given below in Figure 5-

**Figure 4. List of parameter and its processing sequence**

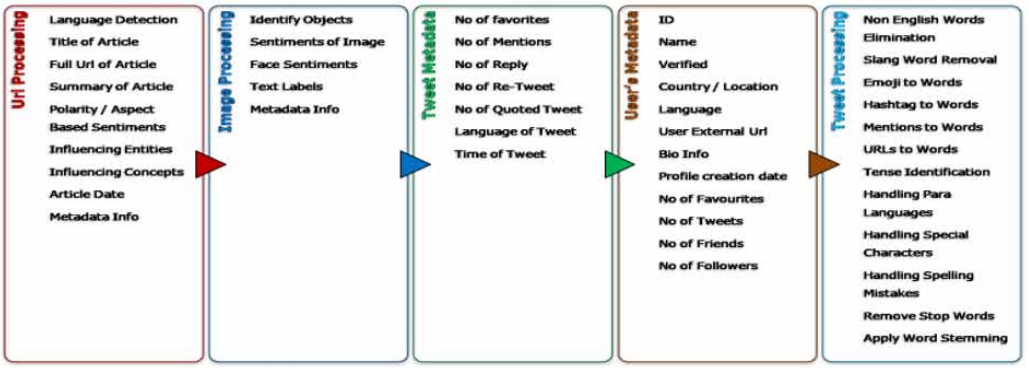
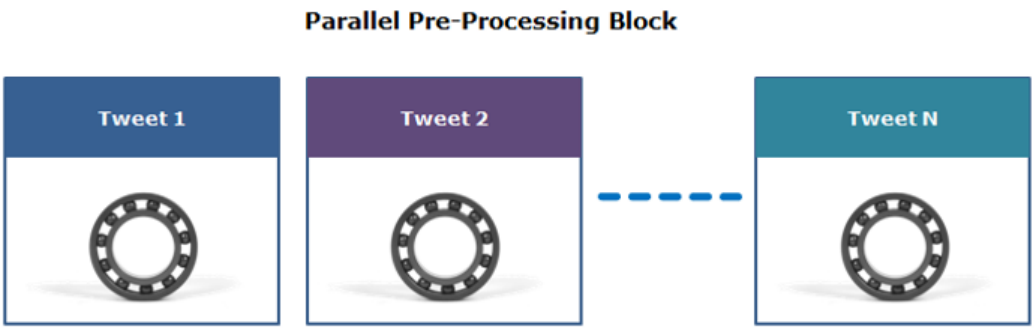



**Figure 5. Horizontally scaled pre-processing block**



#### 3.2.3.2 Third Party API Integration

This framework includes the images, urls and authors profile information too along with post's original content and its metadata, motive behind this inclusion is to regressive use of the framework as well as to obtain the better result. So we are integrated below apis in pre-processing stage to get additional data points-

- Image Processing API
  - AYLIEN
  - Google's computer vision API
  - Clarifai
  - Amazon Rekognition
- URL Processing API
  - AYLIEN text analysis API

### 3.2.4 Database

Databases are required to persist any data for the longer time that could be retrieved by performing queries, there are lots of open source databases available but for this use case we are getting the post data in json format so selection of MongoDB as database would be the best choice as MongoDB is the best candidate for document store NoSQL databases. In this framework MongoDB would be used as permanent storage or for batch processing by leveraging MongoDB's aggregate queries to boost up the performance. MongoDB would also be utilized in cache management along with Redis; all indexing would be available in Redis and as per request Redis fetch the required data in its memory to serve as cache.

#### 3.2.4.1 Cache Management

In this framework Redis is implemented to serve the purpose of cache as well, it is storing the most occurring processed data in its Memory to maximize re-usability as well as to increase the performance. In this framework we are allowing to store top prioritized 100 (that could be customized for more or less data) processed information to serve as cache and rest data would be in MongoDB by keeping their ID in cache, this implementation was done by sorted – list data type of Redis. If records matched with available id but not available in content cache, framework is retrieving the processed data from MongoDB and store in Redis content cache if it is got prioritized. This implementation enables the framework to run with relatively lower memory as well. Effective time to access a record in this cache management system could be calculated with below equation (eq 2)-

$$Time_{effectiveaccess} = Ratio_{Hit} \times Time_{Hit} + Ratio_{Miss} \times Time_{Miss} \quad (2)$$

The representation of cache management is depicted in Figure 6.

**Figure 6. Redis-based cache management**

### *3.2.5 Analytics Block*

This selection and tuning of the model is the most critical and time consuming task, but once it's done, it could be easily deployed on the system to process the incoming data based on the training and tuning. This section is to just concerned to process the data while ignoring the training and tuning of the model. Proposed framework is salable enough to utilize any algorithm with it, but for this research we utilized below machine learning algorithms-

- Naive Bayes
- Maximum Entropy
- Support Vector Machine (SVM)
- Long Short Term Memory (LSTM)

All above mentioned algorithms having different approaches to predict the results on given tagged data to measure the performance and all works well with this framework.

### *3.2.6 Visualization*

Nature of information is being more and more complex, so for better grasping, visualization is very important, in this framework we are going to use matplotlib and python-dash libraries to give the shape to the data, one could choose any of them as per the use case.

- **Matplotlib**: is the most popular open source plotting library for python, it is easy to use and have huge developers support, it could be used to plot some graphs on analytics results.
- **Python-Dash**: is a framework for building analytical web applications with modern UI elements like dropdowns, sliders, and graphs in python, it is MIT licensed open source build on top of Plotly.js, React, and Flask, its premium plan is also available for support and mission critical services.

Ease and effectiveness of visuals to grasp the analysis result could be represented with below equation (eq 3)-

$$Impact_{visuals} = 10 - 25\%_{UnderstandingTime} + 120 - 150\%_{BetterUnderstanding} \quad (3)$$

## 4. EXPERIMENTAL RESULTS

Experimental results of our research would be presented in this section; this framework is implemented for processing social media posts so collection of posts, framework processing environment and results are discussed in subsequent sections.

### 4.1 Experimental Setting

This section will provide detailed information about the data collection strategy and methodology along with the system configuration to set up this framework.

### *4.1.1 Selection of Source*

Proposed framework is built to process user's post, feedback, queries and other types of text input to extract useful insights, so for experimental purpose we chosen Tweets (posts on micro blogging website Twitter), below are the reasons for the selection-

- Global popularity
- Wide range of topics
- Bigger user base (330 million monthly active users out of total 1.3 billion accounts.)
- Users are well distributed by age, gender, languages, countries, education and income ranges.
- Most of the leaders, celebrities, influencers and companies are engaging (Rezapour et al., 2017) there day by day
- Sort text messages with embedded urls & images
- Approx 500 million tweets per day
- Free and premium APIs to collect tweets
- Hashtags (Anjaria & Guddeti, 2014), users and other keywords based search
- Popularity in researcher's world

### *4.1.2 Data Collection Strategy*

After selection of the source we started collecting tweets by using Twitter free API using Tweepy library of python, initially we were storing the tweets in MongoDB directly because twitter free API have limit of 3000 tweets per minute with a max speed of 1K tweets per second (calculation mentioned below based on equation 4 and 5).

At a later stage we started collecting tweets by using multiple Twitter free APIs running parallely for collecting tweets of different categories by search facility from different machines connected in a cloud network so that it couldn't be blocked, and we were able to achieve approx 12K tweets per second where MongoDB was not stable to handle this speed of data on our system configurations. Apart from that twitter free streaming API is providing only 1% of the total data that would not be sufficient to get the accurate sentiments or insights about any product, company or event, so for accurate analysis we used the Twitter Premium API (GNIP API) (Tweepy, ) that would increase the limit to 30K tweets per minute with highest speed of 5K tweets per second. These calculations are based on below equations (eq 4 and eq 5)-

$$MSPM = RPM \times TPRperMinute \tag{4}$$
$$MSPS = RPS \times TPRperSecond \tag{5}$$

Where, MSPM is Max Speed per Minute, MSPS is Max Speed per Second, RPM is Requests per Minute, RPS is Requests per Second and TPR is Tweets per Request.

Calculation for free public API-

- RPM = 30
- TPS = 10
- TPR = 100
- MSPM = 30 x 100 = 3000 per Minute (based on equation 4)
- MSPS = 10 x 100 = 1000 per Second (based on equation 5)

Calculation for premium API-

- RPM = 60
- TPS = 10
- TPR = 500
- MSPM = 60 x 500 = 30000 per Minute (based on equation 4)
- MSPS = 10 x 500 = 5000 per Second (based on equation 5)

At this moment of time it was clear to us that we have to build some framework that could handle this speed of data and proposed frameworks works well for our use case.

#### *4.1.3 Data Set*

Collections of tweets are their analysis was started in 2015 when tweet size was limited to 140 characters, but this research included the higher length of tweets since Nov 2017. This research is based on English texts only, so the topics, hashtags, events and users were considered accordingly.

In this research we considered 7 popular categories of posts and retrieved approx 1 Million of posts in these categories, detailed information are given in Table 1 and its graphical representation is given in Figure 7.

Tagged data is the base of any supervised learning so we did tagging on the collected tweets parallely, these tweets were tagged by different people but each particular category was tagged by a single person just to ensure that personal preferences, skill and biasness could not impact the research. Below are the details of tagging on data in Table 2.

#### *4.1.4 System Configuration*

This research is the part of education program so a variety of systems were utilized in collection, storage and processing of the framework but all the bench-marking and result calculation were done one my personal laptop having mentioned configuration in Table 3.

### 4.2 Experiments

#### *4.2.1 Performance Measure*

In this section we would measure the performance of the proposed framework on single thread and 4 thread systems. Timing, memory utilization and cpu utilization have been considered to measure to the performance of the framework.

##### *4.2.1.1 Activity Wise Performance Benchmarking on Single Thread vs. 4 Threads Parallel Paradigm*

Proposed framework supports parallelism so this section is to present the bench-marking of framework on single thread vs. 4 threads (this bench-marking was done on 4 core system so we chose 4 threads parallelism). In this experiment we splitted the whole processing into multiple sub processes to give more focus on bench-marking on each individual sub processes. Below is the result of bench-marking on 1K tagged tweets in Table 4 along with its graphical representation is given in Figure 8-

**Table 1. Stats of tweet collection used in this research**

| Category | No of Posts | Post Count With Image | Unique Image | Post Count With URLs | Unique URLs | Unique Authors |
|---|---|---|---|---|---|---|
| Technology | 210634 | 48827 | 31144 | 90832 | 39222 | 72142 |
| Politics | 172874 | 73197 | 18389 | 56551 | 11503 | 46891 |
| Nature | 124382 | 89735 | 42161 | 3711 | 2677 | 22802 |
| Sports | 159420 | 49811 | 16006 | 22651 | 9383 | 62051 |
| Entertainments | 92391 | 19562 | 7284 | 25176 | 8604 | 19783 |
| Business | 112794 | 12957 | 6789 | 55702 | 30183 | 23689 |
| Misc. | 139272 | 24178 | 10185 | 13084 | 3029 | 59929 |

**Figure 7. (a) to (g) are displaying the tweet distribution on individual category and (h) is displaying the whole tweet distribution used in this research**


**Table 2. Stats of tagged and untagged data along with training and validation data set size**

| Total | 1011767 |
|---|---|
| Tagged | 50000 |
| Training | 35000 |
| Validation | 15000 |
| Un-Tagged | 961767 |

**Table 3. System configurations details**

| Parameter | Value |
|---|---|
| Memory | 7.5 GiB |
| Processor | Intel® Core™ i5-5200U CPU @ 2.20GHz × 4 |
| Graphics | Intel® HD Graphics 5500 (Broadwell GT2) |
| OS type | 64-bit |
| Disk | 482.1 GB |

**Table 4. Benchmarking in seconds for 1K tagged tweets on single thread vs. 4 thread architecture**

| Activity | Single Thread Benchmarking (in Seconds) | 4 Threads Benchmarking (in Seconds) |
|---|---|---|
| MongoDB Read | 1.023 | 0.563 |
| MongoDB Write | 0.987 | 0.492 |
| Redis Read | 0.495 | 0.281 |
| Redis Write | 0.47 | 0.218 |
| Post Collection | 0.61 | 0.313 |
| Profile Processing | 43.478 | 17.236 |
| Image Processing | 653.532 | 193.712 |
| Urls Processing | 1214.286 | 431.421 |
| Pre-Processing | 829.273 | 254.373 |
| Analytics | 127.624 | 56.135 |
| Visualization | 24.724 | 13.265 |
| Complete Processing | 1023.314 | 424.06 |
| Complete Processing + Cython + Numba | 917.137 | 372.933 |

#### *4.2.1.2 Time, Memory, and CPU Performance Benchmarking*

This bench-marking was done on 961767 un-tagged tweets to measure the effectiveness of the framework on execution time, memory utilization and cpu utilization while processing the tweets. Here we processed the tweets with attached images, urls and author's profile information along with the content of post using this framework. Bench-marking was done on various activity methods along with old architecture, here bench-marking of old architecture was done on processing of only the tweets content not the attached images, urls and author's profile information. Detailed information of the bench-marking are given in Table 5 along with its graphical representation is given in Figure 9.

### *4.2.2 Accuracy Measure*

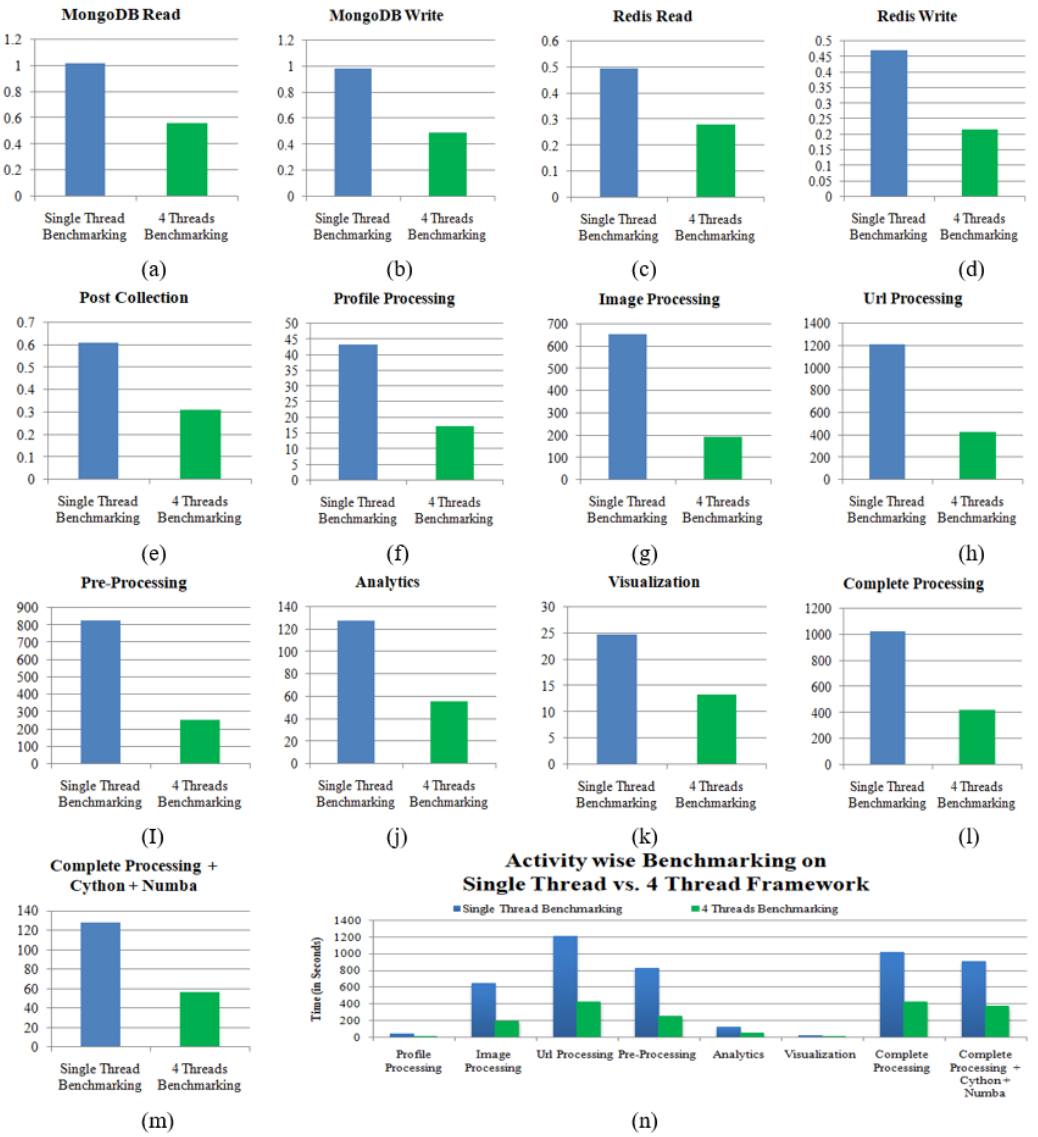
In this section we would measure the accuracy of various machine learning models on different feature groups; apart from that we measured the accuracy of tweets on the basis of their length.

#### *4.2.2.1 Feature Groups vs. Machine Learning Models Accuracy Measure*

In this research we considered content of tweets, images, urls and author profile information as 4 different features so we combined them in various groups to measure their effectiveness in terms of accuracy. While analysis we observed that the feature groups without having content of the posts are not much relevant, so ignore various such combinations and continued our research with below mentioned feature groups-

- Content
- Content + Image
- Content + URL
- Content + Author
- Content + Image + URL
- Content + Image + URL + Author

**Figure 8. (a) to (m) are displaying the bench-marking on each separate process and (n) is displaying the whole bench-marking in one glance**




In this research we utilized below machine learning algorithms having different approaches to predict the results on given tagged data to measure the performance-

- Naive Bayes
- Maximum Entropy
- Support Vector Machine (SVM)
- Long Short Term Memory (LSTM)

This research measured the performance on below mentioned 4 parameters-

**Table 5. Execution timing, memory, and CPU utilization for processing 961767 tweets**

| Activity Methods | Timing (in Hrs) | Memory Utilization (in MB) | CPU Utilization (in %) |
|---|---|---|---|
| Complete Processing | 280.559 | 1837 | 65.13 |
| Complete Processing + Cython + Numba | 244.825 | 1821 | 67.14 |
| Complete Processing (4 Threads) | 118.148 | 6431 | 99.7 |
| Complete Processing (4 Threads) + Cython + Numba | 104.307 | 6352 | 99.99 |
| Complete Processing with Old Architecture | 241.785 | 2104 | 68.122 |

**Figure 9. (a) to (c) are displaying the execution timing in Hrs, memory utilization in MB and CPU utilization in percentage and (d) is displaying the whole bench-marking in one glance**


$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{6}$$

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{9}$$

Where TP, TN, FP and FN extends for true positive, true negative, false positive and false negative respectively.

Result of accuracy measurement of proposed framework are given in Table 6 along with its graphical representation in Figure 10.

#### *4.2.2.2 Tweets Length Wise Accuracy Measure*

While having our analysis we found that results are relatively more accurate on Tweets have length either less that 120 char or more than 140 char, observed reason was less usages of sort form to complete sentence in 140 char limit, below is the analysis on length of tweets using SVM algorithm in Table 7 along with its graphical representation in Figure 11-

**Table 6. Effectiveness measurements of machine learning algorithms for processing 961767 tweets on various feature groups**

| Feature Group | Naive Bayes | | | | Maximum Entropy | | | | SVM | | | | LSTM | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Accuracy | Precision | Recall | F1 | Accuracy | Precision | Recall | F1 | Accuracy | Precision | Recall | F1 | Accuracy | Precision | Recall | F1 |
| Content | 0.66 | 0.66 | 0.66 | 0.66 | 0.74 | 0.73 | 0.74 | 0.73 | 0.81 | 0.80 | 0.81 | 0.80 | 0.82 | 0.81 | 0.82 | 0.81 |
| Content + Image | 0.67 | 0.68 | 0.67 | 0.67 | 0.77 | 0.76 | 0.77 | 0.76 | 0.83 | 0.82 | 0.83 | 0.82 | 0.83 | 0.81 | 0.83 | 0.82 |
| Content + Url | 0.68 | 0.68 | 0.69 | 0.68 | 0.77 | 0.76 | 0.78 | 0.77 | 0.83 | 0.83 | 0.84 | 0.83 | 0.84 | 0.83 | 0.84 | 0.84 |
| Content + Author | 0.66 | 0.66 | 0.66 | 0.66 | 0.75 | 0.74 | 0.75 | 0.74 | 0.82 | 0.81 | 0.82 | 0.81 | 0.82 | 0.81 | 0.82 | 0.81 |
| Content + Image + Url | 0.73 | 0.71 | 0.74 | 0.72 | 0.78 | 0.76 | 0.78 | 0.77 | 0.84 | 0.84 | 0.85 | 0.84 | 0.84 | 0.84 | 0.85 | 0.84 |
| Content + Image + Url + Author | 0.72 | 0.67 | 0.74 | 0.70 | 0.78 | 0.77 | 0.79 | 0.78 | 0.85 | 0.85 | 0.85 | 0.85 | 0.85 | 0.85 | 0.86 | 0.85 |

**Table 7. Effectiveness measurements of machine learning algorithms for processing 961767 tweets on various tweet size**

| Tweet Size | Naive Bayes | | | | Maximum Entropy | | | | SVM | | | | LSTM | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Accuracy | Precision | Recall | F1 | Accuracy | Precision | Recall | F1 | Accuracy | Precision | Recall | F1 | Accuracy | Precision | Recall | F1 |
| < 120 Char Limit | 0.72 | 0.68 | 0.74 | 0.71 | 0.79 | 0.78 | 0.79 | 0.79 | 0.85 | 0.85 | 0.86 | 0.85 | 0.86 | 0.84 | 0.88 | 0.86 |
| 140 Char Limit | 0.71 | 0.67 | 0.72 | 0.70 | 0.77 | 0.76 | 0.77 | 0.77 | 0.83 | 0.82 | 0.83 | 0.82 | 0.83 | 0.82 | 0.83 | 0.83 |
| > 140 Char Limit | 0.73 | 0.70 | 0.74 | 0.72 | 0.80 | 0.79 | 0.80 | 0.80 | 0.86 | 0.85 | 0.86 | 0.86 | 0.87 | 0.86 | 0.88 | 0.87 |

**Figure 10. (a) to (d) are displaying the bench-marking on various feature groups for Naive Bayes, Maximum Entropy, SVM & LSTM respectively and (e) is displaying the whole bench-marking in one glance**

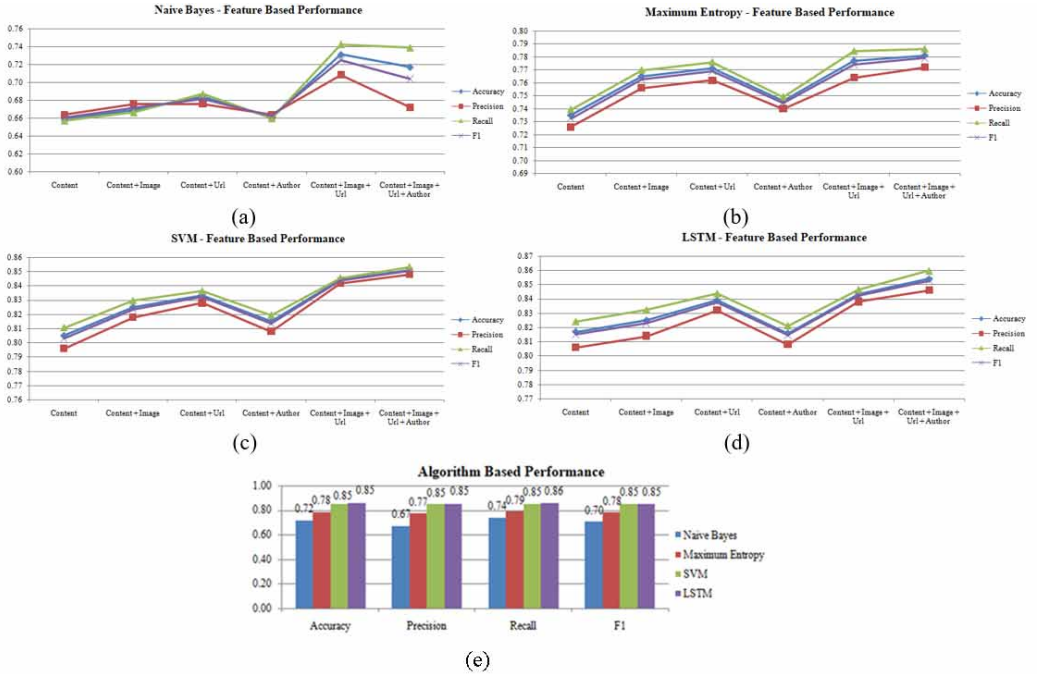
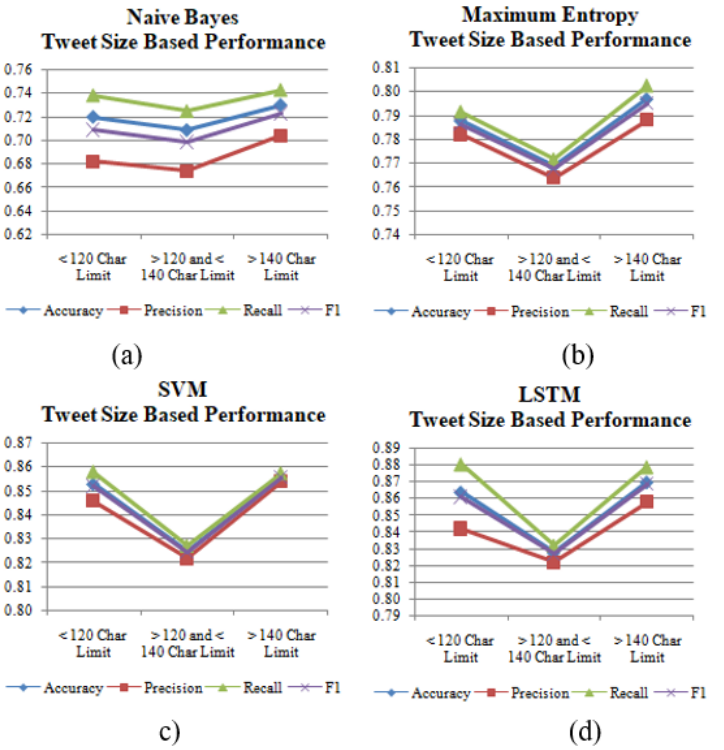



**Figure 11. (a) to (d) are displaying the bench-marking on tweet size for Naive Bayes, maximum entropy, SVM, and LSTM respectively**



## 5. CONCLUSION

This decade is revolutionary in terms of data generation, processing and mining. Big data technologies have gained the foremost attention for horizontal scaling and future expansion of data processing capabilities at the same time it's been over hyped and being utilized on very low throughput systems as well in terms of future expansions, and branding strategies. Beneath the belief that every organization is not processing terabytes of data but still have enough to be processed in single threaded standalone machine, this research represented a simple, fast, sturdy and novel parallel data processing framework for middle tier organizations or personals to achieve the goals of sentiment analysis, trend analysis and so on for its own organization, product, technology, services or personal image while not bearing any additional overhead of big data technologies.

Using 4 threads parallel architecture, this framework achieved 85.1% accuracy in 424.06 seconds for processing 1000 posts (including attached image, urls and authors profile information of the post) while without using this framework we achieved 80.5% accuracy in almost double execution time without processing any image, urls or author's profile information of the post.

## REFERENCES

Anjaria, M., & Guddeti, R. M. R. (2014). Influence factor based opinion mining of twitter data using supervised learning. *6th International Conference on Communication Systems and Networks (COMSNETS)*. IEEE. doi:10.1109/COMSNETS.2014.6734907

Azab, N., & ElSherif, M. (2018). A framework for using data analytics to measure trust in government through the social capital generated over governmental social media platforms. *19th Annual International Conference on Digital Government Research: Governance in the Data Age.* ACM. doi:10.1145/3209281.3209331

Baars, H., & Kemper, H.-G. (2008). Management support with structured and unstructured data – An integrated business intelligence framework. *Information Systems Management*, *25*(2), 132–148. doi:10.1080/10580530801941058

Batrinca, B. (2015). *Social media analytics: a survey of techniques, tools and platforms. AI & Society*. Springer.

Beier, M., & Wagner, K. (2016). Social media adoption: barriers to the strategic use of social media in SMEs. *Proceedings of the european conference on information systems*.

BSA. (2015). *What's the Big Deal With Data?* BSA.

Cao, J., Basoglu, K. A., Sheng, H., & Lowry, P. B. (2015). A systematic review of social networks research in information systems: Building a foundation for exciting future research. *Communications of the Association for Information Systems*, *36*. doi:10.17705/1CAIS.03637

Chebbi, I. (2015). *Wadii Boulila, Imed Riadh Farah, "Big Data: Concepts, Challenges and Applications", Computational Collective Intelligence*. Lecture Notes in Computer Science. Springer.

Fan, W., & Gordon, M. (2014). The Power of Social Media Analytics. *Communications of the ACM*, *57*(6), 74–81. doi:10.1145/2602574

Guellil, I., & Boukhalfa, K. (2015). Social big data mining: A survey focused on opinion mining and sentiments analysis. *12th International symposium on programming and systems*. ISPS. doi:10.1109/ISPS.2015.7244976

Hogenboom, A., Van Iterson, P., Heerschop, B., Frasincar, F., & Kaymak, U. (2011). Determining negation scope and strength in sentiment analysis. *International Conference on Systems, Man, and Cybernetics*. IEEE. doi:10.1109/ICSMC.2011.6084066

IMB. (2012). *Driving marketing effectiveness by managing the blood of big data*. IBM Corp.

Jose, R., & Chooralil, V. S. (2015). Prediction of election result by enhanced sentiment analysis on Twitter data using word sense disambiguation. *International Conference on Control Communication & Computing India (ICCC)*. IEEE. doi:10.1109/ICCC.2015.7432974

Kane, G. C., Alavi, M., Labianca, G., & Borgatti, S. P. (2014). What's different about social media networks? A framework and research agenda. *Management Information Systems Quarterly*, *38*(1), 274–304. doi:10.25300/MISQ/2014/38.1.13

Kaplan, A. M., & Haenlein, M. (2010). Users of the world, unite! The challenges and opportunities of Social Media. *Business Horizons*, *53*(1), 59–68. doi:10.1016/j.bushor.2009.09.003

Karanasou, M., Ampla, A., Doulkeridis, C., & Halkidi, M. (2016). Scalable and Real-time Sentiment Analysis of Twitter Data. *16th International Conference on Data Mining Workshops (ICDMW)*. IEEE. doi:10.1109/ICDMW.2016.0138

Khatua, A., Khatua, A., Ghosh, K., & Chaki, N. (2015). Can# twitter_trends predict election results? Evidence from 2014 indian general election. *48th Hawaii International Conference on System Sciences*. IEEE. doi:10.1109/HICSS.2015.202

Kim, Y., Choi, K. S., & Natali, F. (2016). Extending the Network: The Influence of Offline Friendship on Twitter Network. *Americas Conference on Information Systems*. Rhode Island College.

Kurniawati, K., Shanks, G., & Bekmamedova, N. (2013). The Business Impact of Social Media Analytics. *21st European Conference on Information Systems*. University of Queensland.

Lee, I. (2018). Introduction of Social Media Platforms and Social Media Analytics for Social CRM.. Diverse Methods in Customer Relationship Marketing and Management. IGI Global. doi:10.4018/978-1-5225-5619-0.ch006

Lynn, T., Healy, P., Kilroy, S., Hunt, G., van der Werff, L., Venkatagiri, S., & Morrison, J. (2015). Towards a general research framework for social media research using big data. *International Professional Communication Conference (IPCC).* IEEE. doi:10.1109/IPCC.2015.7235843

Maynard, D., Bontcheva, K., & Rout, D. (2012). Challenges in developing opinion mining tools for social media. *Proceedings of the LREC*. IEEE.

Nulty, P., Theocharis, Y., Popa, S. A., Parnet, O., & Benoit, K. (2016). Social media and political communication in the 2014 elections to the European Parliament,'. *Electoral Studies*, *44*, 429–444. doi:10.1016/j.electstud.2016.04.014

Oh, C., Hu, H.-f., & Yang, W. (2016). Social Media Information Diffusion and Economic Outcomes: Twitter Retweets and Box Office. *Pacific Asia Conference on Information Systems Proceedings*. IEEE.

Oh, C., Sasser, S., & Almahmoud, S. (2015). Social Media Anlaytics Framework: The case of Twitter and Super Bowl Ads. *Journal of Information Technology Management*.

Rezapour, R., Wang, L., Abdar, O., & Diesner, J. (2017). Identifying the overlap between election result and candidates' ranking based on hashtag-enhanced, lexicon-based sentiment analysis. *11th International Conference on Semantic Computing (ICSC).* IEEE. doi:10.1109/ICSC.2017.92

Saha, S., Yadav, J., & Ranjan, P. (2017). Proposed approach for sarcasm detection in twitter. *Indian Journal of Science and Technology*, *10*(25), 1–8. doi:10.17485/ijst/2017/v10i25/114443

Shen, Y., Hock Chuan, C., & Cheng, S. H. (2016). The Medium Matters: Effects on What Consumers Talk about Regarding Movie Trailers. *International Conference on Information Systems*. IEEE.

Stieglitz, S., & Dang-Xuan, L. (2013). Social media and political communication: A social media analytics framework. *Social Network Analysis and Mining*, *3*(4), 1277–1291. doi:10.1007/s13278-012-0079-3

Stieglitz, S., Dang-Xuan, L., Bruns, A., & Neuberger, C. (2014). Social Media Analytics – An Interdisciplinary Approach and Its Implications for Information Systems. *Business & Information Systems Engineering*, *6*(2), 89–96. doi:10.1007/s12599-014-0315-7

Tsantarliotis, P., Pitoura, E. & Tsaparas, (2017). Defining and predicting troll vulnerability in online social media. *P. Soc. Netw. Anal. Min*. IEEE.

Tweepy. (n.d.). *Tweepy: Python Library.* Tweepy: https://www.tweepy.org/

Yadranjiaghdam, B., Yasrobi, S., & Tabrizi, N. (2017). Developing a Real-Time Data Analytics Framework for Twitter Streaming Data. *International Congress on Big Data.* IEEE. doi:10.1109/BigDataCongress.2017.49

Younas, M. (2019). Research challenges of big data. Service Oriented Computing and Applications. Springer.

Zeng, D., Chen, H., Lusch, R., & Li, S. (2010). Social Media Analytics and Intelligence. Intelligent Systems. IEEE.

Zhang, S., Zhao, L., Lu, Y., & Yang, J. (2016). Do you get tired of socializing? An empirical explanation of discontinuous usage behaviour in social network services. *Information & Management*, *53*(7), 904–914. doi:10.1016/j.im.2016.03.006

*Ravindra Kumar Singh is a research scholar pursuing his PhD in Computer Science Department from NIT Jalandhar. He achieved his M.Tech in Computer Science from NIT Jalandhar in 2011 and his research profile includes 15+ international journals and international conferences. Apart from his research activities he is very stable in his professional career and contributed well in various companies, most recently he possessed the role of Technical Architect in Germany based leading health care service provider company powered by AI, ML, Big Data and Blockchain Technologies.*